

PERANCANGAN OTOMATISASI SISTEM PENGISIAN PENAMPUNGAN AIR MENGGUNAKAN METODE PROTOTYPE BERBASIS MIKROKONTROLLER AT89S52 ME

Dian Gustina^{1*}, Yudi Irawan Chandra²

¹ Sistem Informasi, Universitas Persada Indonesia YAI,
Jl. Diponegoro No. 74 Jakarta Pusat

² Sistem Informasi, STMIK Jakarta STI&K,
Jl. BRI Radio Dalam No.17 Kebayoran Baru Jakarta Selatan 12140,
*E-mail : dgustina@yahoo.com

ABSTRAK

Kebutuhan sumber daya alam dan energi untuk manusia pada saat ini sangatlah besar, tetapi jika sumber daya alam dan energi tersebut selalu digunakan terus-menerus secara berlebihan maka dalam jangka waktu dekat sumber daya alam yang ada akan semakin berkurang bahkan habis sama sekali. Program hemat energi sudah banyak disampaikan oleh pemerintah, namun pada kenyataannya di lingkungan bermasyarakat masih banyak terlihat pemborosan dalam menggunakan sumber daya alam dan energi khususnya listrik dan air.

Sehingga perlu adanya upaya untuk mengurangi pemborosan yang terjadi, salah satunya adalah pembuatan sistem yang dapat mengontrol penggunaan listrik dan air di lingkungan sekitar. Pembuatan sistem dibuat secara otomatis, karena pada saat ini banyak kegiatan manusia yang dikerjakan oleh mesin. Oleh karena itu sistem yang digunakan pun memanfaatkan sebuah alat yang terdiri dari komponen-komponen elektronika dengan mikrokontroler AT89S52 sebagai IC terprogram pada alat. Interface juga dibutuhkan sebagai integrasi antara alat dengan komputer, Borland Delphi 7.0 merupakan salah satu bahasa pemrograman tingkat tinggi dengan visualisasi yang mudah dipahami oleh programmer.

Kata Kunci: Otomasi Pengisian Air, Prototype, Mikrokontroler AT89S52 ME

ABSTRACT

Demands on natural resources and energy to humans at this point is great, but if the natural resources and energy is always used continuously excessively so in the near future the natural resources that exist on the wane even exhausted altogether. Energy-saving programs already widely reported by the government, but in reality in the society still widely seen waste in the use of natural resources and energy, particularly electricity and water.

So should the effort to reduce the waste that occurs, one of which is the making of a system that can control the use of electricity and water in the neighborhood. Making the system is automated, because at present many human activities are done by machines. Therefore the system is used also utilizes a device consisting of electronic components to the microcontroller AT89S52 as IC programmed on the tool. Interface is also needed as an integration between tools with computer, Borland Delphi 7.0 is a high level programming language with visuals that easily understood by programmers.

Keywords: Water Filling Automation, Prototype, Microcontroller AT89S52 ME

PENDAHULUAN

Air merupakan salah satu sumber daya alam dan energi yang sangat kita butuhkan untuk bertahan hidup serta sebagai kebutuhan yang paling utama bagi manusia. Faktanya di lingkungan tempat tinggal penulis banyak

sekali warga yang kurang hemat dalam menggunakan air, ketika sebuah rumah melakukan pengisian penampungan air (bak). Apabila penampungan sudah penuh, seringkali terjadi kelalaian yang berakibat air banyak yang terbuang sia-sia. Hal ini dapat merugikan

sebab tidak hanya air saja yang terbuang sia-sia, bahkan listrik yang digunakan juga sangat banyak dan besar.

Untuk menanggulangi pemborosan yang terjadi, penulis berinisiatif untuk membuat sebuah sistem yang dapat mengendalikan/mengontrol penggunaan listrik dan air. Dengan cara membuat sebuah alat yang mudah dan sederhana, sehingga dalam penggunaannya pun tidaklah sulit serta sistem dapat berjalan secara otomatis sesuai dengan apa yang di inginkan dan direncanakan.

Pembuatan alat ini menggunakan Mikrokontroller AT89S52, karena pada AT89S52 adalah salah satu anggota dari keluarga MCS-51/52 yang dilengkapi dengan internal 8 Kbyte Flash PEROM (*Programmable and Erasable Read Only Memory*), yang memungkinkan memori program untuk dapat di program kembali. AT89S52 dirancang oleh Atmel sesuai dengan instruksi standar dan susunan pin 80C5.

Kemudian alat ini di integrasikan dengan komputer, dan sebagai *interface* digunakan *Borland Delphi 7.0*, karena program aplikasi ini merupakan salah satu bahasa pemrograman tingkat tinggi yang mudah dipahami dan perintah yang digunakan dapat dilakukan secara *visual*.

Akibat pemborosan air dan listrik yang berlangsung secara terus-menerus dalam lingkungan penulis, maka untuk mengurangi pemborosan tersebut perlu dilakukan penerapan kesadaran warga untuk berhemat dalam menggunakan sumber daya alam yang ada. Selain itu juga diperlukan sebuah sistem yang dapat mengendalikan pemborosan tersebut.

Dalam hal ini penulis berinisiatif untuk membuat sebuah sistem yang dapat mengendalikan proses pengisian penampungan air sementara tersebut. Jadi apabila penampungan tersebut kosong, maka penampungan air tersebut akan terisi secara otomatis berdasarkan pengaturan (*Setting*) pengguna.

Pada penelitian ini penulis akan membahas tentang otomatisasi sistem pengisian penampungan air (bak) dengan menggunakan Mikrokontroller AT89S52 yang merupakan sebuah IC terprogram yang dapat digunakan untuk mengontrol atau mengendalikan suatu alat. Penulis membatasi masalah tersebut hanya pada pembuatan

rangkaian, *Interface* dengan menggunakan *Borland Delphi 7.0*, dan pembuatan simulasi alat dengan menggunakan bahan *fiber (acrylic)* karena bahan ini sangat mudah didapat.

Adapun tujuan dari penelitian ini adalah membuat alat yang dapat mengendalikan air pada sebuah penampungan, sehingga dapat mengurangi pemborosan listrik dan air yang terbuang sia-sia, sehingga dapat dilakukan penghematan listrik dan air pada tiap rumah.

TINJAUAN PUSTAKA

1. Sistem Pengendalian Otomatis

Sistem adalah susunan, himpunan, atau dapat dikatakan kumpulan benda-benda yang dihubungkan atau saling berhubungan sedemikian rupa sehingga dapat membentuk satu kesatuan. (Jogianto, 1999)

Pengendalian adalah mengarahkan, mengatur, atau memerintahkan secara. Maka sistem pengendalian otomatis adalah susunan komponen-komponen fisik yang dihubungkan sedemikian rupa sehingga dapat mengarahkan, mengontrol, mengatur, atau memerintahkan sistem secara otomatis (tidak secara manual).

Hal yang mempengaruhi sebuah sistem pengendalian adalah adanya masukan (*input*), proses (*process*) serta adanya keluaran (*output*). Komputer merupakan sebuah alat pengendali dalam sebuah sistem, yang dimanfaatkan sebagai masukan (*input*). Sedangkan untuk menghubungkan komputer dengan peralatan lain dibutuhkan sebuah antarmuka (*interface*) agar data yang telah diproses dari masukan dapat menghasilkan sebuah keluaran yang diteruskan ke peralatan lain. *Interface* menggambarkan rangkaian/alat yang dibuat untuk menghubungkan komputer dengan peralatan yang lain.

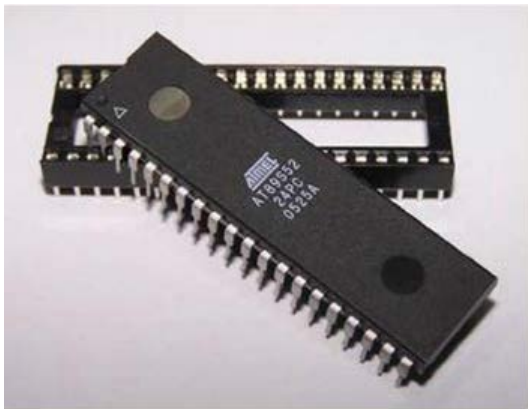
Sistem tidak ada yang tertutup, supaya siklus sistem dapat terus berlangsung maka sistem harus mempunyai sistem pengendalian. Pengendalian suatu sistem dapat berupa pengendalian umpan balik (*feedback control system*), pengendalian umpan maju (*feed forward control system*) dan pengendalian pencegahan (*preventive control system*). (Jogiyanto Hartono, 1999)

2. Mikrokontroller AT89S52

Mikrokontroller AT89S52 adalah sebuah mikrokontroller 8 bit bertegangan rendah dengan teknologi CMOS berkinerja tinggi yang dilengkapi memori flash yang

dapat di program sebesar 8 Kbyte. Komponen ini dibuat dengan teknologi memori Atmel yang *nonvolatile* dan berkapasitas tinggi serta *compatible* dengan set instruksi dan pin out standar industri 80CSI. Flash on-chip memungkinkan memori program diprogram ulang dalam sistem atau oleh pemrograman memori *non-volatile* yang konvensional. Dengan menggunakan CPU 8 bit dengan flash yang diprogram dari sistem dalam sebuah *monolithic chip*.

AT89S52 adalah salah satu anggota dari keluarga MCS-51/52 yang dilengkapi dengan internal 8 Kbyte Flash PEROM (*Programmable and Erasable Read Only Memory*), yang memungkinkan memori program untuk dapat diprogram kembali. AT89S52 dirancang oleh Atmel sesuai dengan instruksi standar dan susunan pin 80C5. (Much. Ibnu Malik, 2005)



Gambar 1. Mikrokontroler AT89S52

Mikrokontroler AT89S52 memiliki :

1. Sebuah CPU (*Central Processing Unit*) 8 Bit.
2. 256 byte RAM (*Random Acces Memory*) internal.
3. Empat buah port I/O, yang masing masing terdiri dari 8 bit
4. Osilator internal dan rangkaian pewaktu.
5. Dua buah *timer/counter* 16 bit
6. Lima buah jalur interupsi (2 buah interupsi eksternal dan 3 interupsi internal).
7. Sebuah port serial dengan *full duplex* UART (*Universal Asynchronous Receiver Transmitter*).
8. Mampu melaksanakan proses perkalian, pembagian, dan Boolean.
9. EPROM yang besarnya 8 KByte untuk memori program.

10. Kecepatan maksimum pelaksanaan instruksi per siklus adalah 0,5 μ s pada frekuensi *clock* 24 MHz. Apabila frekuensi *clock* mikrokontroler yang digunakan adalah 12 MHz, maka kecepatan pelaksanaan instruksi adalah 1 μ s

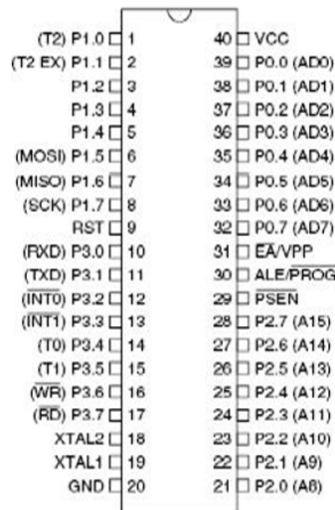
CPU berfungsi mengendalikan seluruh operasi pada mikrokontroler. Unit ini terbagi atas dua bagian, yaitu unit pengendali atau CU (*Control Unit*) dan unit aritmatika dan logika atau ALU (*Aritmetic logic Unit*) Fungsi utama unit pengendali adalah mengambil instruksi dari memori (*fetch*) kemudian menterjemahkan susunan instruksi tersebut menjadi kumpulan proses kerja sederhana (*decode*), dan melaksanakan urutan instruksi sesuai dengan langkah-langkah yang telah ditentukan program (*execute*). Unit aritmatika dan logika merupakan bagian yang berurusan dengan operasi aritmatika seperti penjumlahan, pengurangan, serta manipulasi data secara logika seperti operasi AND, OR, dan perbandingan. (Usman, 2008)

Bagian Masukan/Keluaran (I/O) berfungsi sebagai alat komunikasi serpih tunggal dengan piranti di luar sistem. Sesuai dengan namanya, perangkat I/O dapat menerima maupun memberi data dari /ke serpih tunggal.

Ada dua macam piranti I/O yang digunakan, yaitu piranti untuk hubungan serial UART (*Universal Asynchronous Receiver Transmitter*) dan piranti untuk hubungan paralel yang disebut dengan PIO (*Pararel Input Output*). Kedua jenis I/O tersebut telah tersedia di dalam serpih tunggal AT89S52. (Usman, 2008)

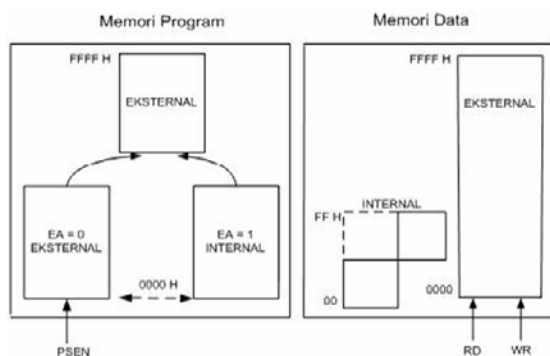
Serpih tunggal keluarga MCS-51 memiliki bahasa pemrograman khusus yang tidak dipahami oleh jenis serpih tunggal yang lain. Bahasa pemrograman ini dikenal dengan nama bahasa *assembler* yang memiliki 256 perangkat instruksi. Namun saat ini pemrograman mikrokontroler dapat dilakukan dengan menggunakan bahasa C. Dengan bahasa C, pemrograman mikrokontroler menjadi lebih mudah, hal ini karena dengan format bahasa C akan secara otomatis diubah menjadi bahasa *assembler* dengan format file hexa

Mikrokontroler AT89S52 mempunyai 40 pin dengan catu daya tunggal 5 Volt. Ke-40 pin tersebut digambarkan sebagai berikut :



Gambar 2. Pin-pin AT89S52

Semua serpih tunggal dalam keluarga MCS-51 memiliki pembagian ruang alamat untuk program dan data. Pemisahan memori program dan memori data memperbolehkan memori data untuk diakses oleh alamat 8bit. Sekalipun demikian, alamat data memori 16 bit dapat dihasilkan melalui register DPTR (*Data Point Register*). Memori program hanya bisa dibaca tidak bisa ditulis karena disimpan dalam EPROM. Dalam hal ini EPROM yang tersedia di dalam serpih tunggal AT89S52 sebesar 8 Kbyte.



Gambar 3. Memori Program dan Memori Data

Mikrokontroler AT89S52 memiliki osilator internal yang dapat digunakan sebagai *clock* bagi CPU. Untuk menggunakan osilator internal, diperlukan sebuah Kristal atau *resonator* keramik antara pena XTAL 1 dan XTAL 2 (pin 19 dan pin 18) yang akan dipasangkan dengan kapasitor yang dihubungkan ke *ground*. Untuk jangka frekuensi operasi mikrokontroler AT89S52 adalah 6 Mhz sampai dengan 24 Mhz, sedangkan untuk pasangan kapasitornya dapat

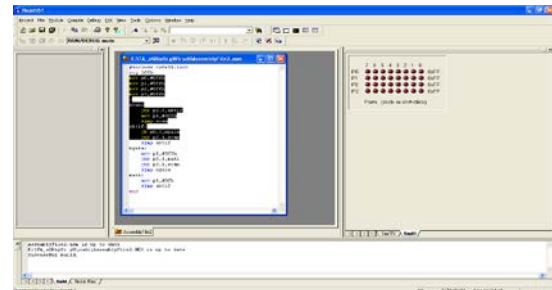
digunakan kapasitor keramik yang bernilai antara 27 pF sampai dengan 33 pF. (Eko Putra, 2009)



Gambar 4. Bentuk dan Simbol Kristal

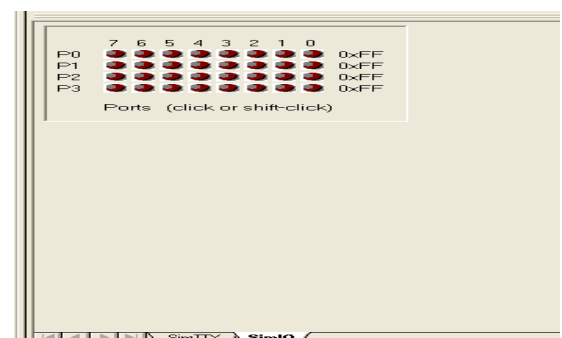
3. Reads51

Reads51 adalah salah satu software pembentuk bahasa mesin. Program yang dituliskan pada Reads51 ini, nantinya akan diubah menjadi file dengan *extension hex* (.Hex). file *hex* inilah yang akan dimasukkan pada mikrokontroler AT89S52. Tampilan Reads51 dapat dilihat pada gambar berikut ini :



Gambar 5. Tampilan Reads51

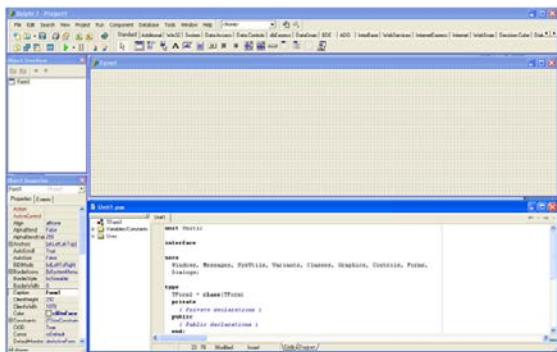
Pada Reads51 program yang dibuat menggunakan bahasa *Assembly* dalam bentuk file *asm*. Jika program telah selesai, maka program dapat di *debugging* untuk mengetahui apakah program yang kita jalankan sudah *success* atau belum



Gambar 6. SimIO Port pada Reads51

4. Borland Delphi 7.0

Merupakan program aplikasi *database* yang berbasis Object Pascal dari Borland. Selain itu, Delphi juga memberikan fasilitas pembuatan aplikasi visual. Delphi merupakan pilihan dalam pembuatan aplikasi visual karena memberikan produktifitas yang tinggi. Delphi 7 memberikan fasilitas untuk dua *platform*, yaitu untuk *platform Windows* dan *Linux*. Delphi untuk *Linux* sebelumnya dikemas dalam sebuah aplikasi terpisah yang bernama *Kylix*, tetapi Delphi 7 menyatukannya dalam sebuah aplikasi. *Library* untuk *Windows* disebut VCL dan *Library* untuk *Linux* disebut CLX.



Gambar 7. Tampilan Borland Delphi

5. Teknik Prototype

Seringkali seorang customer sulit menentukan input yang lebih terinci, proses yang diinginkan dan output yang diharapkan. Tentu saja ini menyebabkan developer tidak yakin dengan efisiensi algoritma yang dibuatnya, sulit menyesuaikan sistem operasi, serta interaksi manusia dan mesin yang harus diambil. Dalam hal seperti ini, pendekatan *prototype* untuk *software engineering* merupakan langkah yang terbaik. *Prototype sebenarnya adalah suatu proses yang memungkinkan developer membuat sebuah model software*.

Ada 2 bentuk dari model ini, yaitu :

1. Paper Prototype

Menggambaran interaksi manusia dan mesin dalam sebuah bentuk yang memungkinkan user mengerti bagaimana interaksi itu terjadi.

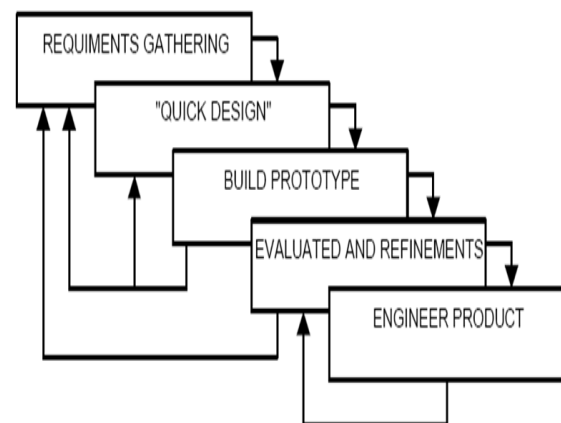
2. Working Prototype

Adalah *prototype* yang mengimplementasikan beberapa bagian dari fungsi *software* yang diinginkan

seperti pada pendekatan pengembangan *software*. Model ini dimulai dengan :

- Pengumpulan kebutuhan developer dan customer
- Menentukan semua tujuan *software*
- Mengidentifikasi kebutuhan-kebutuhan yang diketahui

Hasil dari pengumpulan kebutuhan diteruskan pada *Quick Design*. *Quick Design* ini memfokuskan pada representasi aspek-aspek *software* yang dapat dilihat oleh user, misalnya format input dan output, selanjutnya dari desain cepat diteruskan pada pembentukan *prototype* (langkah ke 3). *Prototype* ini dievaluasi oleh customer / user dan digunakan untuk memperbaiki kebutuhan-kebutuhan *software*. Proses iterasi terjadi agar *prototype* yang dihasilkan memenuhi kebutuhan customer, juga pada saat yang sama developer mengerti lebih baik tentang apa yang harus dikerjakan.



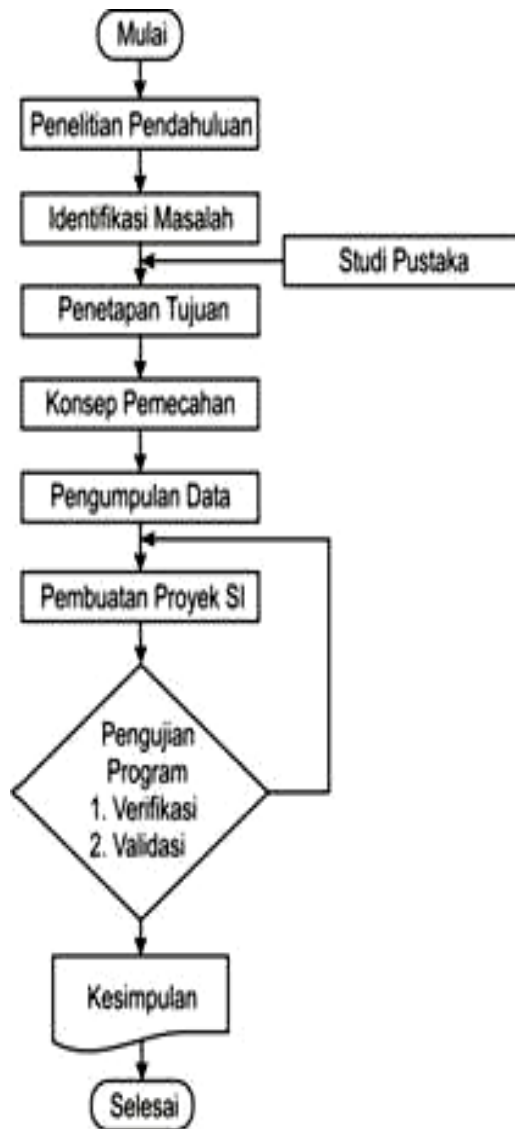
Gambar 8. Metode Prototype

Masalah yang dihadapi oleh *prototyping* paradigma ini adalah :

- a. Customer hanya melihat pada apa yang dihasilkan oleh *software*, tidak peduli pada hal-hal yang berhubungan dengan kualitas *software* dan pemeliharaan jangka panjang.
- b. Developer seringkali menyetujui apa yang diterangkan oleh customer agar *prototype* dapat dihasilkan dengan cepat. Akibatnya timbul pemilihan sistem operasi / bahasa pemrograman yang tidak tepat.

METODE

Skema bagan alir dalam tahapan penelitian tentang pembuatan aplikasi ini dapat dilihat pada gambar berikut :



Gambar 9. Skema Metode Penelitian

Adapun teknik yang dilakukan untuk pengumpulan data adalah sebagai berikut :

1. Metode observasi lapangan

Metode pengumpulan data dengan cara mengamati secara langsung tentang kegiatan, keadaan umum, dan kejadian-kejadian yang ada dalam objek penelitian dengan pencatatan secara otomatis, selain itu metode ini juga dapat dilakukan dengan cara mengadakan tanya jawab langsung dengan masalah yang diteliti bersama narasumber yang dapat dipercaya.

2. Metode Kajian Pustaka

Metode pengumpulan data yang dapat diperoleh melalui perpustakaan atau nara sumber buku lain untuk memperoleh data tambahan yang berhubungan dengan penelitian

HASIL DAN PEMBAHASAN

Cara kerja program pada otomatisasi sistem pengisian penampungan air ini merupakan cara kerja dari program mikrokontroller AT89S52. Program akan menscan apakah pada *port 0.0* dan *port 0.1* ada inputan, jika ada inputan maka program akan menghasilkan *output* pada *port 2.0* dan *port 2.1*. Program akan selalu berjalan (*looping*) selama program belum dimatikan (OFF).

1. Inisialisasi Mikrokontroller

Inisialisasi nilai awal pada mikrokontroller AT89S52 pada *port0*, *port1*, *port2* dan *port3* dengan nilai FF hexadecimal (OFFh). Hal ini dimaksudkan agar semua *portnya* berlogika *high* pada saat program mulai dijalankan, inisialisasi dari program tersebut adalah :

```

MOV P0, #0FFh
MOV P1, #0FFh
MOV P2, #0FFh
MOV P3, #0FFh

```

2. Pengaktifan Pompa Air

Pendeteksian *input* terdiri dari 2 kondisi, yaitu kondisi bak pada saat penuh dan kondisi bak pada saat kosong. Jika kondisi air menyentuh sensor (A) pada *port0.0*, maka penampungan (bak) penuh. Sedangkan jika kondisi air menyentuh sensor (B) pada *port0.1*, maka penampungan (bak) kosong. *Syntax* untuk pengaktifan pompa air adalah sebagai berikut:

Scan :

```

JNB P0.0, aktif
MOV P1, Scan
SJMP Scan

```

Aktif :

```

JB P0.0, nyala
JNB P2.1, scan
SJMP aktif

```

Nyala :

```

MOV P1, #0F0h
JNB P0.1, Mati
JNB P2.1, scan
SJMP nyala

```

3. Penon-aktifan Pompa Air

Penon-aktifan pompa air ini dilakukan apabila penampungan (bak) air tidak ingin diisi kembali, jadi secara keseluruhan pompa akan mati dan jika ingin di aktifkan kembali maka harus di pilih *menu* ON pada aplikasi. *Syntax* untuk menon-aktifkan pompa air adalah sebagai berikut :

```
Mati :  
MOV P1, #0Fh  
Sjmp aktif  
End
```

4. Pembuatan Simulasi Alat

Pada proses pembuatan simulasi alat ini, penulis menggunakan bahan *acrylic (fiber)* seperti pada gambar berikut :



Gambar 10. Bentuk Simulasi Alat

Simulasi alat dibuat dari bahan *acrylic*, karena bahan ini sangat sesuai dengan konsep otomatisasi sistem pengisian penampungan air. Karena pada simulasi alat, *indicator* yang digunakan adalah air. Sehingga bahan *acrylic* sangat tepat untuk menampung debit air tersebut.

Pada simulasi alat terdapat satu buah kotak pada bagian atas dan dua buah penampungan (bak) pada bagian bawah, fungsi dari bentuk simulasi alat tersebut adalah:

1. Kotak bagian atas sebagai tempat papan PCB, hal ini bertujuan agar rangkaian alat tidak basah. Karena di khawatirkan apabila rangkaian ini terkena air, maka dapat merusak rangkaian tersebut.
2. Bak pertama (atas) berfungsi sebagai penampungan air (bak) yang telah dipasang *sensor water level indicator*. Bak bagian atas ini terdiri dari ukuran panjang = 19 cm, Lebar = 10 cm, Tinggi = 5,5 cm, dan Volume = 1045 cm³.
3. Bak kedua (bawah) berfungsi sebagai tempat menampung air dari bak pertama apabila air pada bak pertama dikosongkan. Pada bak ini diletakkan pompa air, karena pada saat proses pengisian bak pertama pompa akan menyedot dan mengalirkan (mengisi) air ke bak pertama melalui selang yang dipasang pada pompa. Bak bagian bawah ini terdiri dari ukuran panjang = 22 cm, Lebar = 13 cm, Tinggi = 9,5 cm, dan Volume = 2717 cm³.

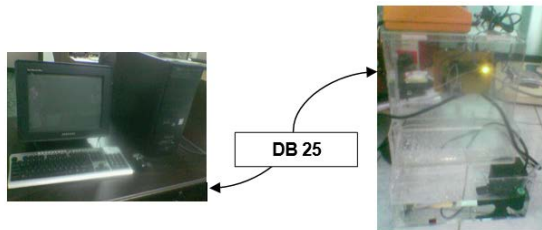
5. Testing (Percobaan) Simulasi Alat

Setelah melewati tahapan-tahapan proses :

1. Perancangan rangkaian komponen-komponen elektronika.
2. Persiapan komponen-komponen elektronika.
3. Pembuatan rangkaian (implementasi) ke papan PCB.
4. Pembuatan *interface* dengan *Borland Delphi 7.0*.

Yang merupakan urutan kegiatan yang dilakukan dalam pembuatan otomatisasi sistem pengisian penampungan air seperti pada pembahasan bab 3 sebelumnya, kemudian dilanjutkan dengan pembuatan simulasi alat dari bahan *acrylic*.

Setelah Proses pembuatan simulasi alat, maka proses selanjutnya adalah integrasi antara komputer dengan simulasi alat. Proses ini diawali dengan penginputan program *interface Borland Delphi 7.0* dalam bentuk *exe*, jika program *exe* telah di *copy* ke komputer kemudian komputer langsung dihubungkan dengan alat melalui DB 25.



Gambar 11. Integrasi Komputer dengan Alat

Dari gambar di atas, dapat dilihat proses integrasi antara komputer dengan simulasi alat. Sedangkan *interface* di *copy* ke komputer, karena pada *interface Borland Delphi 7.0* program sudah dalam bentuk *exe* sehingga program *Delphi* dapat langsung di *Run* (dijalankan).

Pada percobaan (*testing*) simulasi alat, dilakukan apabila alat telah terhubung dengan komputer melalui DB 25 yang ada pada *acrylic*.

Selanjutnya penampungan (bak) bawah diisi air, kemudian jalankan program aplikasi dengan menekan tombol ON yang ada pada *exe* seperti gambar berikut :



Gambar 12. Tampilan Aplikasi Program

Pada saat menjalankan aplikasi *Delphi* ketika tombol (ON) maka pompa air yang diletakkan di bak bawah akan langsung menyedot air dan mengalirkannya ke bak atas melalui selang yang telah terhubung dengan pompa dan bak atas. Kemudian pada aplikasi *Delphi* akan tercetak “PROSES MENGISI BAK” sebagai peringatan bahwa bak atas dalam keadaan mulai diisi air. Apabila bak telah penuh (air menyentuh *high level*), maka akan tercetak peringatan pada aplikasi “BAK PENUH” mengartikan bahwa kondisi bak dalam keadaan penuh. Jika air di dalam bak telah berkurang (tidak menyentuh *high level*) lagi, maka akan tercetak peringatan “STANDBY” karena aplikasi menunggu sampai air menyentuh *low level* kembali. Proses seperti ini akan selalu berulang sampai di klik tombol OFF, karena tombol OFF ini

digunakan untuk mematikan alat sedangkan untuk keluar dari aplikasi klik tombol EXIT.

Tabel 1. Tabel Hasil Uji Coba Alat

NO	Sensor	Percobaan	Kondisi	Hasil	Relay
1	PCB dengan Lempengan Biasa	Tombol ON	"Proses Mengisi Bak"	Berhasil	Bunyi
2	PCB dengan Lempengan Biasa	Bak Penuh	"Bak Sudah Penuh" kemudian kondisi "Alat Standby"	Tidak Berhasil	Tidak Bunyi
3	PCB dengan Lempengan Biasa	Pengosongan Bak Atas	"Alat Standby"	Tidak Berhasil	-
4	PCB dengan Lempengan Biasa	Bak Kosong	"Proses Mengisi Bak" kembali	Tidak Berhasil	Tidak Bunyi
5	PCB dengan Lempengan Biasa	Tombol OFF	OFF (alat mati)	Berhasil	Bunyi
6	PCB dengan Lempengan Biasa	Tombol EXIT	Keluar dari aplikasi <i>Delphi</i>	Berhasil	-
7	PCB dengan Resistor	Tombol ON	"Proses Mengisi Bak"	Berhasil	Bunyi
8	PCB dengan Resistor	Bak Penuh	"Bak Sudah Penuh" kemudian kondisi "Alat Standby"	Berhasil	Bunyi
9	PCB dengan Resistor	Pengosongan Bak Atas	"Alat Standby" kemudian "Proses Mengisi Bak" kembali	Berhasil	-
10	PCB dengan Resistor	Bak Kosong	"Proses Mengisi Bak" kembali	Berhasil	Bunyi
11	PCB dengan Resistor	Tombol OFF	OFF (alat mati)	Berhasil	Bunyi
12	PCB dengan Resistor	Tombol EXIT	Keluar dari aplikasi <i>Delphi</i>	Berhasil	-

Pada uji coba alat tahap awal, sensor yang digunakan hanya berupa PCB dengan lempengan kuningan. Dan hasilnya proses pengisian bak tidak bekerja secara optimal, jika bak kosong pompa tidak mengisi bak kembali. Maka sensor diganti menggunakan PCB dengan resistor, agar sensor tidak terlalu *sensitive* terhadap air. Resistor menghambat arus yang mengalir apabila air menyentuh resistor. Sehingga alat dapat bekerja secara optimal.

SIMPULAN DAN SARAN

Dari hasil pembuatan otomatisasi sistem pengisian penampungan air dengan menggunakan *Reads51* sebagai program pada mikrokontroler AT89S52 dengan bahasa *assembly* dan *Borland Delphi 7.0* sebagai *interface* antara alat dengan *Personal Computer*. Maka dapat diambil kesimpulan otomatisasi sistem pengisian penampungan air, antara lain:

1. Membantu mengontrol dan mengurangi pemborosan penggunaan listrik dan air dalam mengisi bak.
2. Memudahkan pengguna, karena komponen dari alat yang digunakan sangat sederhana dan mudah di dapatkan.

3. Tahap paling awal dalam pembuatan alat ini adalah perancangan rangkaian dari sistem yang akan digunakan. Kemudian implementasi ke papan PCB, selanjutnya pembuatan program dengan menggunakan Reads51. Program tersebut akan dimasukkan ke Mikrokontroller AT89S52 dengan cara di *download* menggunakan *prolog*, dan pembuatan *interface Delphi 7.0* dilanjutkan dengan pembuatan simulasi alat. Tahap akhir adalah *testing* (integrasi) alat dengan *interface*, apakah alat bekerja secara sesuai dengan apa yang diharapkan yaitu otomatisasi sistem pengisian penampungan air.

Setelah menjalani proses penelitian dan pembahasan, maka Penulis ingin memberikan saran-sarpenulisaan sebagai berikut:

1. Daya yang dibutuhkan untuk simulasi alat cukup kecil, tetapi apabila ingin digunakan dengan daya yang lebih besar lagi maka komponen-komponen elektronika pada alat juga harus ditambah lagi daya dan tegangannya.
2. Jika kapasitas daya listrik yang dibutuhkan besar, alat ini tidak dapat bekerja secara optimal.
3. *Sensor* yang digunakan jangan terlalu *sensitive* dengan air, jika terlalu *sensitive* maka alat tidak bekerja dengan baik sesuai dengan program.
4. Pembuatan sistem ini hanya sebatas otomatisasi pengisian penampungan (bak) air saja pada saat bak kosong, tetapi untuk pengaturan daya listrik yang digunakan tidak ada.

DAFTAR PUSTAKA

- Artanto, 2009, Dian, *Merakit PLC Dengan Mikrokontroller*, Jakarta, Elek Media Komputindo.
- Eko Putra, 2009, Agfianto, *Belajar Mikrokontroller AT89C51/52/55*, Jakarta, Gava Media.
- Hartono, Jogyanto, 1999, *Pemrograman dan Sistem Informasi dan Intelegensi Buatan* : Yogyakarta, Andi Yogyakarta.
- Jogianto, Andi, *Aplikasi Cerdas Menggunakan Delphi*, Jakarta, Andi Publisher, 2003
- Kurniawan, Rudy, 2009, *Penulisan Ilmiah Pendeteksi Ketinggian Air Sebagai Peringatan Banjir*, Bekasi.
- Madcoms, 2006, *Seri Panduan Pemrograman – pemrograman Borland Delphi 7 Lengkap Dengan Contoh Aplikasi*, Jakarta, Andi Offse
- Malik, Moh. Ibnu, 2005, *Belajar Mikrokontroller ATMEL AT89S52*, Jakarta, Gaya.
- M. Husen, Nano, 2008, *Tulisan Ilmiah Rancang Bangun Sistem Kran Otomatis Menggunakan Sensor PIR*, Bekasi.
- Pressman RS. 2010. *Software Engineering : A Practitioner's Approach, 7th ed.*Mc Grow Hill.
- Usman, 2008, *Teknik Antarmuka + Pemrograman Mikrokontroller AT89S52*, Jakarta, Andi Publisher.